

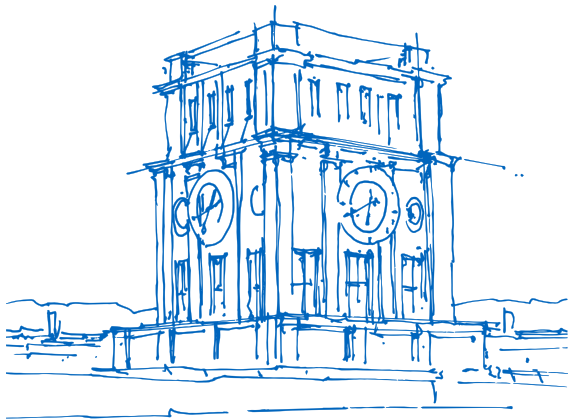
Übung 10: Parallelisierung

Einführung in die Rechnerarchitektur

Michael Morandell

School of Computation, Information and Technology
Technische Universität München

06. – 12. Januar 2025



Montags:

<https://zulip.in.tum.de/#narrow/stream/2668-ERA-Tutorium—Mo-1000-4>



Donnerstags:

<https://zulip.in.tum.de/#narrow/stream/2657-ERA-Tutorium—Do-1200-2>



Website: <https://home.in.tum.de/momi/era/>

Keine Garantie für die Richtigkeit der Tutorfolien.
Bei Unklarheiten/Unstimmigkeiten haben VL/ZÜ-Folien recht!

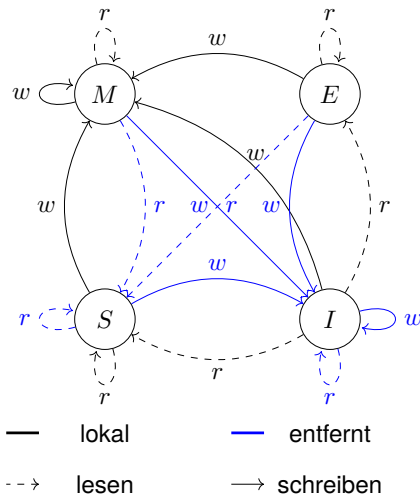
- Quiz
- Wiederholung
- Tutorblatt
 - ☐ Speedup
 - ☐ Roofline Modell
 - ☐ Auswertung von Speedupdiagrammen

- Single-Threaded Rechenleistung immer weiter durch physikalische Limits eingeschränkt
- Optimierungen: Pipelining, Out-of-Order-Processing, Ausnutzen von Parallelität
- SIMD: Eine Instruktion, die gleichzeitig auf mehrere Daten ausgeführt wird (mehr dazu in GRA)

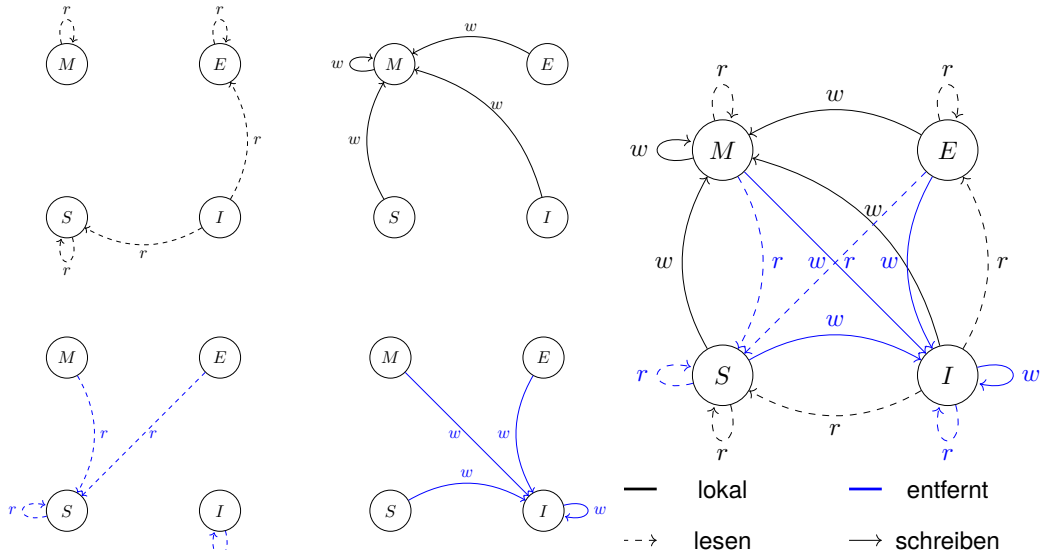
		Instruction stream	
		Single	Multiple
Data stream	Single	SISD	MISD
	Multiple	SIMD	MIMD

Quelle: [A Taxonomy of Reconfigurable Single-/Multiprocessor Systems-on-Chip](#)

- Mehrkernsysteme: Was wenn CPU1 und CPU2 beide ein Datum gecached haben und es modifizieren?
→ Cache-Inkonsistenzen
- Einführung von Zuständen für Cachezeilen
- CPUs hören jeweils die Zugriffe der anderen Kerne ab („Bus Snooping“)
- **M**odified, (**E**xclusive), **S**hared, **I**nvalid
- Exclusive-Bit ermöglicht kleineren Overhead wenn CPUs auf verschiedenen Cache-Blöcken arbeiten



MESI: Übersicht



Mit t_s sequentieller Programmteil, t_p paralleler Programmteil, n Anzahl CPU-Kerne, T Ausführungszeit mit $n = 1$:

- Amdahlsches Gesetz: Gleiche Problemgröße, aufgeteilt auf mehrere Kerne \rightarrow begrenzt durch sequentiellen Anteil

$$S_{\text{Amdahl}}(n) = \frac{T}{t_s + \frac{t_p}{n}}$$

- Gustafsons Gesetz: Mehr Kerne können mehr berechnen: Größeres Problem \rightarrow paralleler Anteil wächst mit Problemgröße, t_s proportional kleiner

$$S_{\text{Gustafson}}(n) = \frac{t_s + n \cdot t_p}{T}$$

- Zwei verschiedene Perspektiven, abhängig von Problemszenario verschieden geeignet

$$p = 1$$

$t_{\text{sequential}}$	t_{parallel}
-------------------------	-----------------------

$$p = 3$$

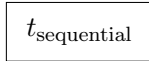
Quelle: Vorlesungsmaterial ERA

Amdahlsches Gesetz

$p = 1$



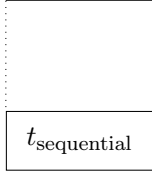
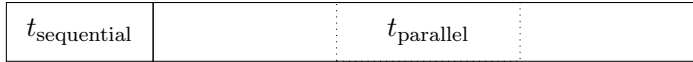
$p = 3$



Quelle: Vorlesungsmaterial ERA

Amdahlsches Gesetz

$p = 1$



$p = 3$

Quelle: Vorlesungsmaterial ERA

Amdahlsches Gesetz

$p = 1$

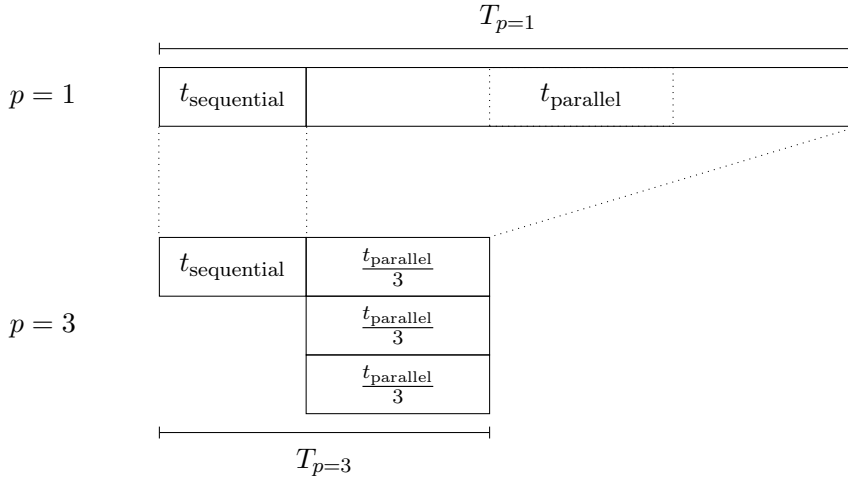
$t_{\text{sequential}}$		t_{parallel}	
-------------------------	--	-----------------------	--

$p = 3$

$t_{\text{sequential}}$	$\frac{t_{\text{parallel}}}{3}$
	$\frac{t_{\text{parallel}}}{3}$
	$\frac{t_{\text{parallel}}}{3}$

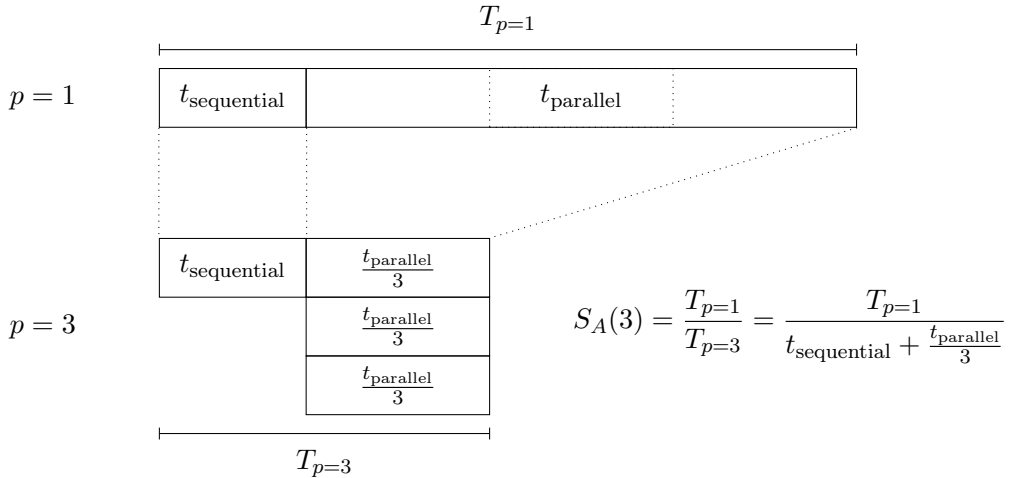
Quelle: Vorlesungsmaterial ERA

Amdahlsches Gesetz



Quelle: Vorlesungsmaterial ERA

Amdahlsches Gesetz



Quelle: Vorlesungsmaterial ERA

$$p = 1$$

$t_{\text{sequential}}$	t_{parallel}
-------------------------	-----------------------

$$p = 3$$

$p = 1$

$t_{\text{sequential}}$	t_{parallel}
-------------------------	-----------------------

--	--

$t_{\text{sequential}}$	t_{parallel}
-------------------------	-----------------------

$p = 3$

$p = 1$

$t_{\text{sequential}}$	t_{parallel}
-------------------------	-----------------------

--	--

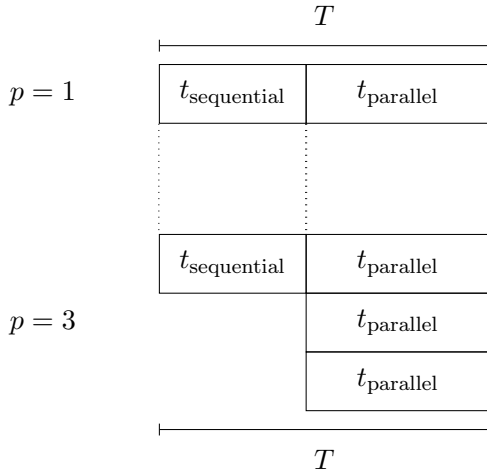
$p = 3$

$t_{\text{sequential}}$	t_{parallel}
-------------------------	-----------------------

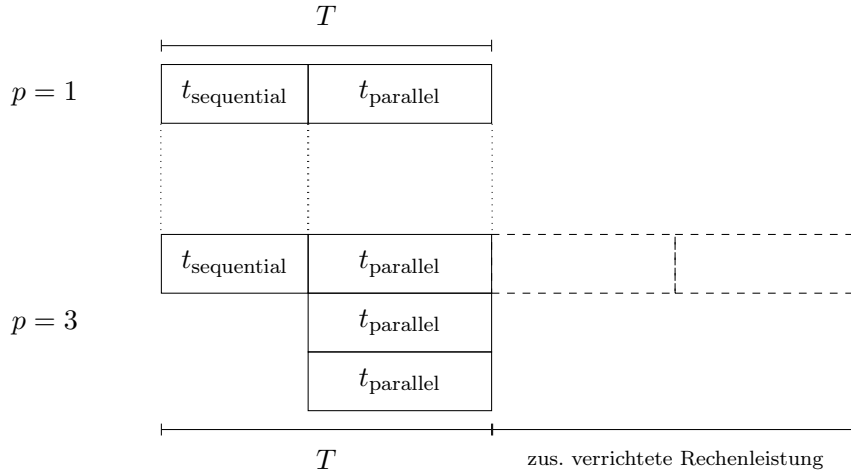
t_{parallel}

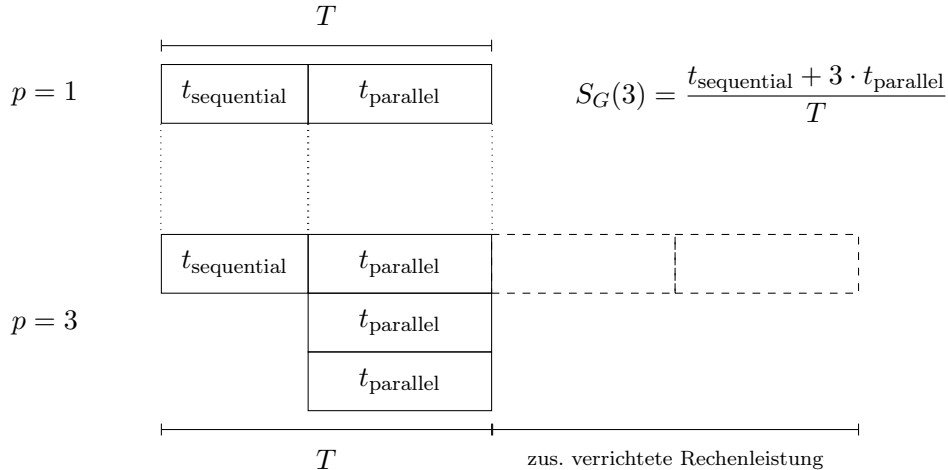
t_{parallel}

Gustafsons Gesetz



Gustafsons Gesetz

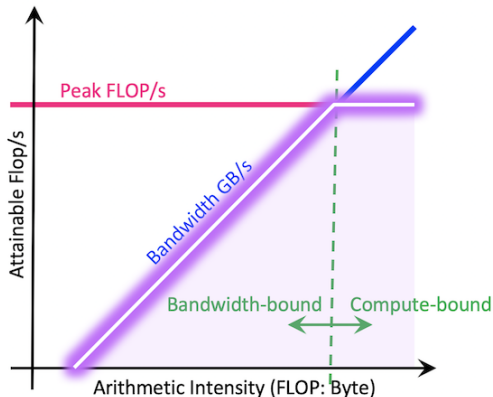




- Floating Point Operations per Second
- Maß für die Leistungsfähigkeit von Computern oder Prozessoren
- Beispiele:
 - (CPU) Pentium 4 3,2 GHz: Spitzenleistung 6,4 GFLOPS
 - (CPU) Intel Core i7 5820k, 6K/12T, 3,3 GHz: Spitzenleistung 273,1 GFLOPS
 - (Supercomputer) SuperMUC NG: theoretische Spitzenleistung 26,8739 PFLOPS

Roofline-Modell

- Modellierung der maximalen Gesamtleistung
- X-Achse: Arithmetische Intensität (I)
 - in FLOP pro Byte
 - $I = (\text{Anzahl Rechenoperationen in FLOP}) / (\text{Speichertransfers in Byte})$
 - Abhängig vom Algorithmus
- Y-Achse: Rechenleistung
 - in FLOPS
 - Abhängig vom Rechensystem
- Diagonale: Max Speicherbandbreite
 - $y/x = (\text{GFlop/s}) / (\text{Flop/Byte}) = \text{GB/s}$



Quelle



<https://tinyurl.com/era-tut>

Ein Teil der Folien stammt aus dem Foliensatz von Niklas Ladurner. Vielen Dank dafür!